

REMARKS

Reconsideration and allowance of the above-referenced application are respectfully requested. Claims 23, 28, and 34 are canceled, new claims 40-42 are added, and claims 1-22, 24-27, 29-33, and 35-42 are pending in the application.

Further to the Request for Corrected Office Action mailed March 10, 2006, Applicant presumes that claims 2-11, 15-18, 20-29 and 31-39 recite allowable subject matter, based on the absence of any art-based rejection of these claims.

Claims 1, 12-14, 19 and 30 stand rejected under 35 USC §102(e) in view of U.S. Patent Publication No. 2002/0055980 to Goddard. These rejections are respectfully traversed.

Each of the independent claims 1, 12, 19 and 30 specify emulation in a *protocol emulator*, where IP frames are promiscuously detected on a network interface. *An executable emulation application* within the protocol emulator generates, *for each corresponding detected IP frame*, a response IP frame. Each response IP frame is output by a raw socket onto the network interface.

As described in the specification (e.g., page 2, lines 17-23, page 2, line 28 to page 3, line 3), the claimed protocol emulator is able to emulate an unlimited number of IP addresses based on scalable operations, such as promiscuous detection of IP frames on the network interface that eliminates the necessity for conventional IP filtering of received IP frames that normally is performed by a UNIX kernel. Moreover, the generation of a response IP frame *for each corresponding detected IP frame* results in the executable emulation application being able to emulate an unlimited number of IP devices, since a UNIX descriptor is no longer needed for each and every IP address emulated by the protocol emulator; rather, the executable emulation application generates a response IP frame for *each detected IP frame*, regardless of IP address. Finally, each response IP frame is output by the raw socket onto the network interface, enabling kernel resources to be bypassed, further reducing the reliance on operating system resources such as the UNIX kernel.

These and other features are neither disclosed nor suggested in the applied prior art.

Goddard provides no disclosure or suggestion of any protocol emulation, as claimed. Rather, Goddard describes a dispatcher (e.g., 102 of Fig. 1, 212 of Fig. 2, para. 28-31) that receives client requests from client devices, and selectively forwards the requests to a back-end server (e.g., 104 of Fig. 1, 202/204 of Fig. 2) that executes the client requests. In fact, Goddard emphasizes that “the performance of a server may be enhanced by limiting the amount of data processed by that server at any given time” (para. 33).

Further, even though Goddard describes in para. 55-56 a “promiscuous mode”, each packet received by the dispatcher *still* undergoes filtering by the dispatcher:

[0055] When a packet arrives at the datalink layer of the dispatcher 210, ***the packet is preferably applied to each filter defined by the dispatcher***, as shown in FIG. 5. The packet capture device then captures all the packets in which it is interested. For example, the packet capture device can operate in a promiscuous mode, during which all packets arriving at the datalink layer are copied to ***a packet capture buffer and then filtered, through software, according to, e.g., their source IP or MAC address, protocol type, etc. Matching packets can then be forwarded*** to the application making the packet capture call, whereas ***non-matching packets can be discarded***. Alternatively, packets arriving at the datalink layer can be filtered through hardware (e.g., via a network interface card) in addition to or instead of software filtering. ...

Hence, Goddard explicitly requires packet filtering to be performed *before* the packet is sent to the application layer, even if packets are received in promiscuous mode, and neither discloses nor suggests generating, ***for each corresponding (promiscuously) detected IP frame***, a response IP frame by an executable emulation application, as claimed. Moreover, Goddard specifically avoids generating a response IP frame for ***each promiscuously detected IP frame*** by (1) filtering the promiscuously detected packets based on MAC addresses or IP addresses, *and* (2) selectively dropping packets that do not meet the prescribed criteria (e.g., as specified in para. 55), or denying client service requests due to overload conditions (e.g., para. 37, lines 3-9; para. 43, 45).

As apparent from the foregoing, Goddard is incapable of generating a response IP frame ***for each corresponding (promiscuously) detected IP frame***, because Goddard requires filtering of incoming packets by the dispatcher in order to prevent overloading of the the back-end server;

in contrast, the claimed executable emulation application is able to generate a response for each corresponding detected IP frame.

Hence, the §102 rejection should be withdrawn because it fails to demonstrate that the applied reference discloses each and every element of the claim. As specified in MPEP §2131: “A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference’ *Verdegaal Bros. V. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). ... ‘The identical invention must be shown in as complete detail as is contained in the ... claim.’ *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).” MPEP 2131 (Rev. 3, Aug. 2005, at p. 2100-76).

For these and other reasons, the rejection of claims 1, 12, 19, and 30 should be withdrawn.

The dependent claims are believed in allowable form in view of the foregoing.

In view of the above, it is believed this application is in condition for allowance, and such as Notice is respectfully solicited.

To the extent necessary, Applicant petitions for an extension of time under 37 C.F.R. 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a), to Deposit Account No. 50-1130, under Order No. 95-451, and please credit any excess fees to such deposit account.

Respectfully submitted,



Leon R. Turkevich
Registration No. 34,035

Customer No. 23164
Date: August 7, 2006